Steady State Lid Driven Cavity Using Simple Algorithm - Finite Volume Method

Ganesh Borde

December 2024

Abstract

This project focuses on the numerical simulation of flow inside a two-dimensional lid-driven cavity. The study investigates the flow properties at low Reynolds numbers to achieve a steady-state solution. A pressure correction approach, specifically the Semi-Implicit Method for Pressure Linked Equations (SIMPLE), is employed to solve the incompressible Navier-Stokes equations. The finite volume discretization (FVM) method is used. The numerical results are validated against benchmark solutions of cavity flow reported in the literature. Various flow characteristics and results are analyzed in detail.

1 Introduction

The lid-driven cavity is a benchmark problem in the study of Computational Fluid Dynamics (CFD). Its simplicity allows researchers to evaluate the accuracy and stability of numerical methods. This problem has been extensively studied for over 50 years. Let us discuss about the problem below.

1.1 Problem description

The lid-driven cavity is a two-dimensional domain bounded by three stationary walls and one moving boundary, i.e., the top wall moving with a velocity U, as shown in Fig. **??**. The cavity is considered a square domain, where the length of all sides is L. As the top wall moves with a velocity U, the flow inside the cavity experiences motion due to the viscous nature of the fluid.

The velocity values at the three fixed boundaries are $u = v = 0$, while the moving boundary has $u = U$ and $v = 0$. In this project, the cavity is studied by setting the velocity $U = 1\,\mathrm{m/s}$ and the length of the boundary $L = 1\,\mathrm{m}$. Only the viscosity of the flow is varied, while the density of the flow is kept constant. The flow properties are analyzed for different viscosities. The pressure fields are not included in the results section, as their sole purpose in this study is to project the velocity field onto the solenoidal space.

1.2 Governing equations

As mentioned, the flow is steady, incompressible, two-dimensional and isothermal. Therefore, the 2-D incompressible Navier-Stokes equations are considered, which are divided into the continuity equation, x-momentum equation, and y-momentum equation.

After non-dimensioning the variables as:

- Length scale $x = \overline{x}L$ and $y = \overline{y}L$

- Velocity scale $u = \overline{u}U$ and $v = \overline{v}U$

- Pressure $p = \overline{p}\rho U^2$

The resulting governing equations (omitting the bar notations for the sake of simplicity) are:

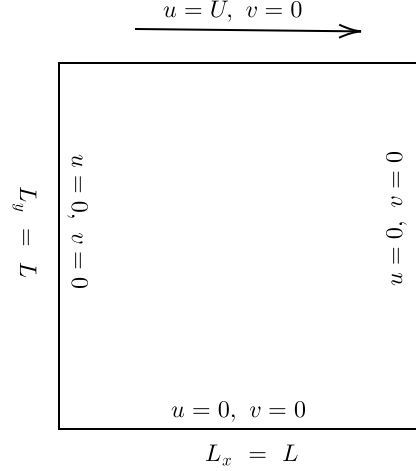$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{1.1}$$

Figure 1.1: Problem description of lid-driven cavity flow.

$$\frac{\partial uu}{\partial x} + \frac{\partial uv}{\partial y} = -\frac{dp}{dx} + \frac{1}{\text{Re}}\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right], \tag{1.2}$$

$$\frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y} = -\frac{dp}{dy} + \frac{1}{\text{Re}}\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right]. \tag{1.3}$$

In Eq. 1.2 and 1.3, Re is reynolds number of the flow, defined as:

$$\text{Re} = \frac{\rho U L}{\mu}, \tag{1.4}$$

where μ is the viscosity of the fluid.

1.3 Literature review

Lid-driven cavity flow problem is a test case type problem for incompressible solvers. The past work done on lid-driven cavity flow gives a range for steady solution. For a wide range of Reynolds numbers, Shen [**?**] examined the flow structures in a unit cavity and discovered that the flow establishes a steady character for Re 10000.

The difference between the projection method and the SIMPLE method is the projection method is generally second-order accurate, which is developed for the simulation of incompressible unsteady flows by employing a non-linear update of pressure term, which may depend on the grid size, time step, and even velocity. It has three and four-step projection method. One of the famous methods by Chorin .[**?**].The standard SIMPLE method is written in a concise formula for steady and unsteady flow. It is proven that SIMPLE type methods have second-order temporal accuracy for unsteady flows. The classical second-order projection method and SIMPLE type methods are united within the framework of the general second-order projection formula [**?**].

2 Numerical method

This section deals with the numerical method, discretization, and grid approach in solving the lid-driven cavity. The SIMPLE algorithm on the staggered grid configuration is used. First, let's talk about the staggered grid.
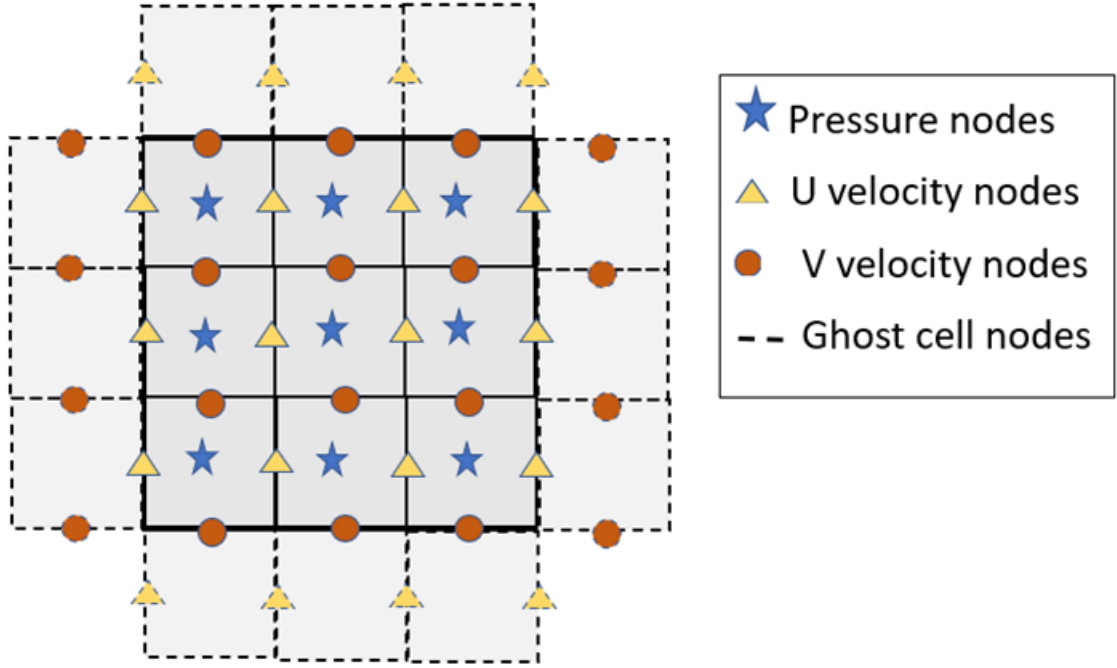
Figure 2.1: Staggered grid configuration

Table 1: The number of variables in the domain

Field quantity	Interior resolution	Resolution with boundary points
p	$n_x \times n_y$	$(n_x + 2) \times (n_y + 2)$
u	$(n_x - 1) \times n_y$	$(n_x + 1) \times (n_y + 2)$
v	$n_x \times (n_y - 1)$	$(n_x + 2) \times (n_y + 1)$

2.1 Staggered grid

A staggered grid is a configuration used for spatial discretization. Scalar variables (such as pressure p, density ρ, total enthalpy h, etc.) are stored at the cell centers of control volumes, whereas velocity components (such as u and v) or momentum variables are stored at the cell faces. This configuration differs from a collocated grid, where all variables are stored at the same locations. For compressible or incompressible flow simulations using structured grids, staggered storage is primarily utilized.

Collocated grids are prone to a discretization error known as odd-even decoupling, which leads to checkerboard oscillation patterns in the solutions. By using a staggered grid, odd-even decoupling between pressure and velocity can be effectively avoided. In a staggered grid, pressure values (p) are located at the center of the cells, while the velocity components u and v are stored at the cell faces, as shown in Fig. **??**.

Boundary conditions can be defined in two different forms: Dirichlet boundary condition, where primary variables are specified, and Neumann boundary condition, where derivatives, such as pressure gradients, are specified.

To apply boundary conditions, ghost cells (cells beyond the actual computational domain) are introduced. Ghost cells in the x-direction are represented by dashed lines, which indicate the v-velocity, while those in the y-direction represent the u-velocity. Pressure nodes are considered in both the x- and y- directions. The updated number of variables in the system is presented in Table 1.

2.2 Discretization

The governing equations are discretized using finite volume method. The velocity at cell center p, is approximated by cell centers of east e, west w, north n, and south s. The u and v are indexing at (i, j) (they are not on same point, because they have meshes for each other) and p at (I, J). So, the x momentum equation (Eq. 1.2) can be approximated as

$$\frac{(u\hat{u})_e - (u\hat{u})_w}{\Delta x} + \frac{(u\hat{v})_n - (u\hat{v})_s}{\Delta y} = -\frac{p_{I+1,j} - p_{I-1,j}}{\Delta x} + \frac{1}{Re}\left[\frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{(\Delta x)^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j=1}}{(\Delta y)^2}\right],$$
$$(2.1)$$

where the u, v velocities at each cell centers as

$$v_n = \frac{v_{i,j} + v_{i+1,j}}{2}, v_s = \frac{v_{i,j-1} + v_{i+1,j-1}}{2}, \tag{2.2}$$

and

$$u_e = \frac{u_{i,j} + u_{i+1,j}}{2}, u_w = \frac{u_{i,j} + u_{i-1,j}}{2}, u_n = \frac{u_{i,j} + u_{i,j+1}}{2}, u_s = \frac{u_{i,j} + u_{i,j-1}}{2}. \tag{2.3}$$

In Eq. 2.1 Δx and Δy is the grid spacing in x and y direction respectively. This equation can be written as systems of equations as

$$a_p u_{i,j} = a_e u_{i+1,j} + a_w u_{i-1,j} + a_n u_{i,j+1} + a_s u_{i,j-1} - \frac{p_{I+1,J} - p_{I,J}}{\Delta x} \tag{2.4}$$

where the coeffients

$$a_e = \frac{1}{Re(\Delta x)^2} - \frac{\hat{u}_e}{2\Delta x}, a_w = \frac{1}{Re(\Delta x)^2} - \frac{\hat{u}_w}{2\Delta x}, a_n = \frac{1}{Re(\Delta y)^2} - \frac{\hat{v}_n}{2\Delta y}, a_s = \frac{1}{Re(\Delta y)^2} - \frac{\hat{v}_s}{2\Delta y}, \tag{2.5}$$

where $\hat{u}_e, \hat{u}_w, \hat{v}_n, and\hat{v}_s$ are velocities can be approximated by using upwind schemes or linear interpolations, but in this study the linear interpolations are used.

Similarly, the y momentum equation can be written as

$$a_p v_{i,j} = a_e v_{i+1,j} + a_w v_{i-1,j} + a_n v_{i,j+1} + a_s v_{i,j-1} - \frac{(p_{I,J+1} - p_{I,J-1})}{\Delta y}, \tag{2.6}$$

where the coefficients are same as defined in Eq. 2.5, but the u, v velocities at each cell centers as approximated differently compared to Eq. 2.2 and 2.3 , which are

$$u_e = \frac{u_{i,j} + u_{i,j+1}}{2}, u_w = \frac{u_{i-1,j} + u_{i-1,j+1}}{2}, \tag{2.7}$$

and

$$v_n = \frac{v_{i,j} + v_{i,j+1}}{2}, v_s = \frac{v_{i,j} + v_{i,j-1}}{2}, v_e = \frac{v_{i,j} + u_{i+1,j}}{2}, v_w = \frac{v_{i,j} + u_{i-1,j}}{2}. \tag{2.8}$$

2.3 SIMPLE

The SIMPLE algorithm used in this study is described below:

- Setup the grid using the staggered grid configuration.

- Initialize the variables u, v, and p with an initial guess.

- Using the initial guess, compute the intermediate velocities u^* and v^* using the momentum equations i.e. Eq. 2.4 and 2.6.

- Solving for pressure p using some approximation, consider

$$u = u^* + u', v = v^* + v', p = p^* + p' \tag{2.9}$$

where u' and v' are correction velocities, solving for this using momentum equation gives

$$a_p u'_{i,j} = a_e u'_{i+1,j} + a_w u'_{i-1,j} + a_n u'_{i,j+1} + a_s u'_{i,j-1} - \frac{p'_{I+1,J} - p'_{I,J}}{\Delta x}, \qquad (2.10)$$

$$a_p v'_{i,j} = a_e v'_{i+1,j} + a_w v'_{i-1,j} + a_n v'_{i,j+1} + a_s v'_{i,j-1} - \frac{(p'_{I,J+1} - p'_{I,J-1})}{\Delta y}, \qquad (2.11)$$

in the SIMPLE method the contribution of correction from neighboring nodes is neglected, other method like SIMPLER, and SIMPLEC consider them. Which results the correction velocities as

$$u'_{i,j} = -\frac{1}{a_p}\frac{p'_{I+1,J} - p'_{I,J}}{\Delta x}, v'_{i,j} = -\frac{1}{a_p}\frac{(p'_{I,J+1} - p'_{I,J-1})}{\Delta y} \qquad (2.12)$$

for simplicity $-\frac{1}{a_p \Delta x}$ and $-\frac{1}{a_p \Delta y}$ are written as d_u and d_v respectively. a_p is different for u and v velocities, from now they are written as a_p^u and a_p^v. Substituting Eq. 2.12 in Eq. 2.9 gives

$$u_{i,j} = u^*_{i,j} + d_u(p'_{I+1,J} - p'_{I,J}), v_{i,j} = v^*_{i,j} + d_v(p'_{I,J+1} - p'_{I,J-1}). \qquad (2.13)$$

Substituting Eq. 2.13 in the continuity equation Eq.1.1 gives a poisson equation to solve for correction pressure as

$$A_P p'_{I,J} = A_E p'_{I+1,J} + A_W p'_{I-1,J} + A_N p'_{I,J+1} + A_S p'_{I,J-1} + b_{I,J}, \qquad (2.14)$$

where the coefficients are

$$A_E = \frac{1}{a^u_{p,(i,j)}}, A_W = \frac{1}{a^u_{p,(i-1,j)}}, A_N = \frac{1}{a^v_{p,(i,j)}}, A_s = \frac{1}{a^u_{p,(i,j-1)}}, A_P = A_E + A_W + A_N + A_S, \qquad (2.15)$$

and

$$b_{i,j} = \frac{u^*_{i,j} - u^*_{i-1,j}}{\Delta x} + \frac{v^*_{i,j} - v^*_{i,j-1}}{\Delta y} \qquad (2.16)$$

- Update the pressure p^{n+1} by summing the initial pressure p and the correction pressure p', with introducing a under relaxation factor α_p as

$$p^{n+1} = p + \alpha_p p' \qquad (2.17)$$

- Calculate the correction velocities u' and v' from the correction pressure as given in Eq. 2.13.

- Update the final velocities u^{n+1} and v^{n+1} for the next time step using a under relaxation factor α_u and α_v as:

$$u^{n+1} = u^* + \alpha_u u', \quad v^{n+1} = v^* + \alpha_v v'.$$

- Assign the updated values u^{n+1}, v^{n+1}, and p^{n+1} to the initial variables u, v, and p for the next iteration.

- Repeat the loop until the solution until the solution converges.

- Finally, transform the velocities from the staggered grid to the collocated grid for plotting results.

3 Results

The lid driven cavity problem has been solved for different Reynolds numbers using the SIMPLE method. The results for low Reynolds numbers 100 and 400 have been computed, for Re 1000 the solver was showing some diverging results. So, the post processing results have been done for Re 100 and 400 with grid size as 129×129 nodes. The problem has a steady state solution, so choosing the tolerance rate effects the result of the system. The tolerance of 1×10^{-6} have been chosen based on the execution time and solution. The SIMPLE algorithm is solved using MATLAB for computations and post processing.
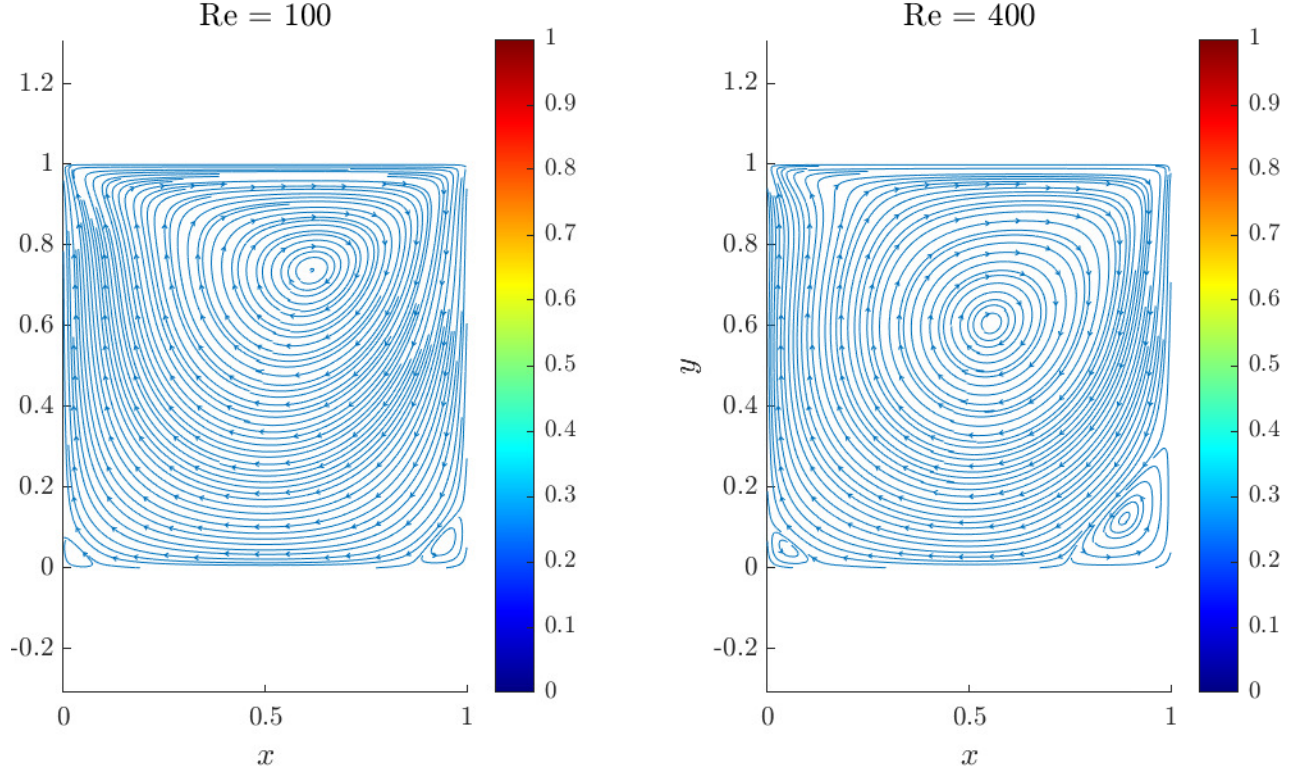
Figure 3.1: v Velocity Contours

3.1 Comparison of velocities

The velocity values for Re 100 and 400 and 129×129 node points from the post processing are plotted along the axis and they are compared with results obtained from Ghia et al. [?] on the same plot. u velocity is plotted along a vertical bisector for both Re 100 and 400, which is shown in Fig. ??. v velocity is plotted along the horizontal bisector for both Re 100 and 400, which is shown in Fig. ??.

The slope towards the wall becomes sharper as the Reynolds number rises. The explanation is that while the viscous force predominates at low Reynolds numbers, it swiftly diminishes as numbers rise. As a result, in flows with high Reynolds numbers, the convective force dominates, and the viscous force is quite weak.

3.2 Comparison of the streamlines

The post processing from MATLAB gives the streamline tracking, the streamline for the Re 100 and 400 are shown in Fig. ??nd Fig. ??espectively. They are compared with streamline from Ghia et al. [?]. The result from SIMPLE algorithm and Ghia et al. looks identical. As the Reynolds increases from 100 to 400 the streamlines look different, where the secondary vortices appear in corners of cavity, they are rotating in the opposite direction. The primary vortices density decreases, and they tend to move to center as Reynolds number increases the secondary vortices grow.

3.3 Convergence

The residual error verses the number of iterations is plotted in post processing which is shown in Fig. ?? for Re 100 and 400. The execution time and number of iterations executed for convergence are shown in Table ??.
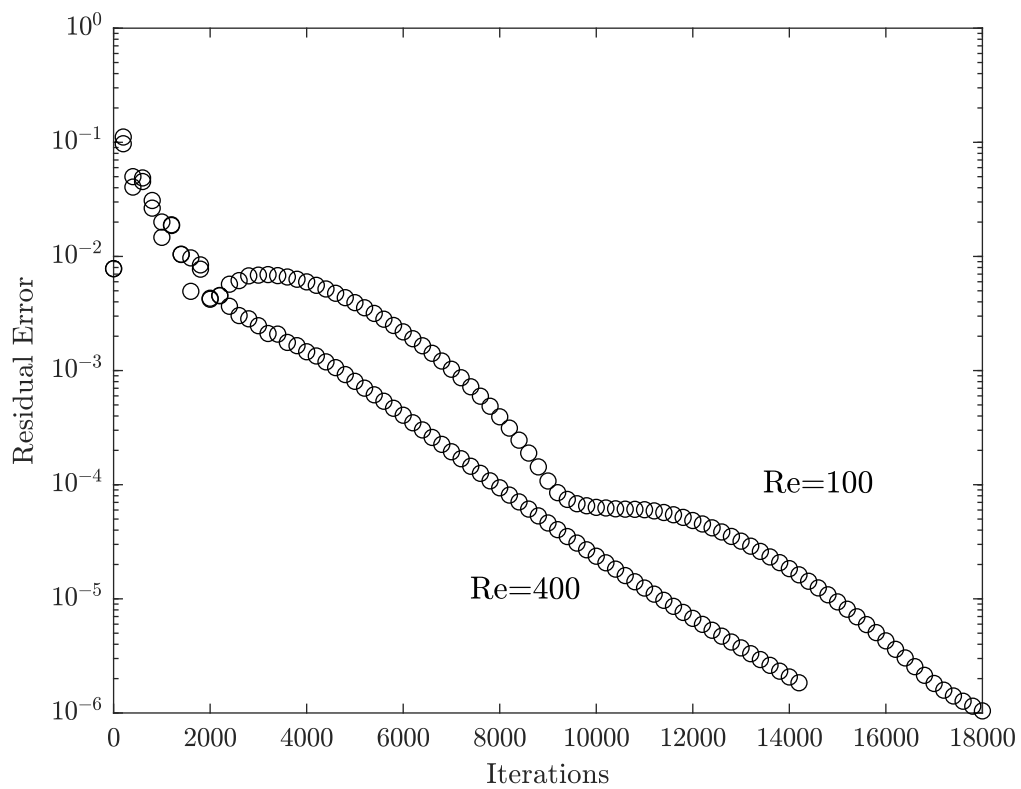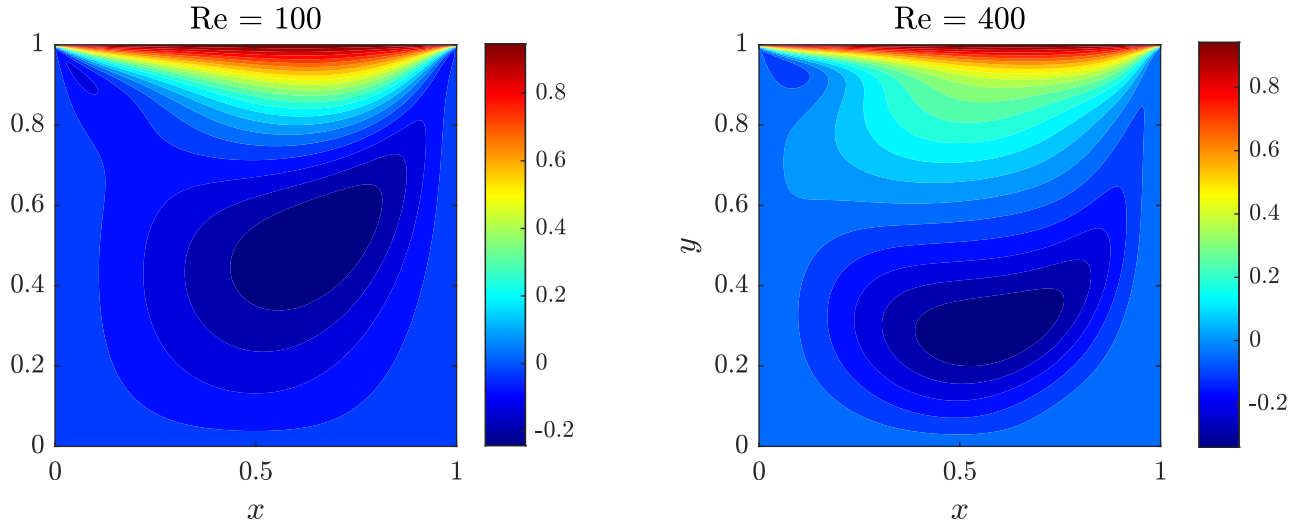
Figure 3.2: v Velocity Contours

Figure 3.3: u Velocity Contours

3.4 Contours

4 Conclusion

In this project, the lid-driven cavity flow is solved using the SIMPLE method with varying Reynolds numbers. The velocity and streamline results are validated against the reference results from Ghia et al. [**?**]. Grid size plays a crucial role in solution convergence and accuracy. Solving incompressible Navier-Stokes equations involves numerous intermediate steps, each of which must be computed carefully.

While solving the momentum equations, the convection term is discretized using the upwind method for velocity. Other methods, such as the QUICK scheme and the power-law scheme, offer higher accuracy and could be considered for further improvement. The Navier-Stokes momentum equations are solved implicitly, enabling larger time steps and potentially faster convergence rates. Solver performance and accuracy can be enhanced by employing these advanced methods. The time required to solve the pressure Poisson equation can be reduced by using sparse matrices and applying the Gauss elimination method to solve the pressure field efficiently.

In summary, the lid-driven cavity flow serves as a benchmark test case for evaluating new computational fluid dynamics techniques due to its simple geometry and flow physics. In the future, I plan to work on solving the lid-driven cavity flow using the Fractional-Step Projection method, an implicit solver for incompressible and time-dependent Navier-Stokes equations.
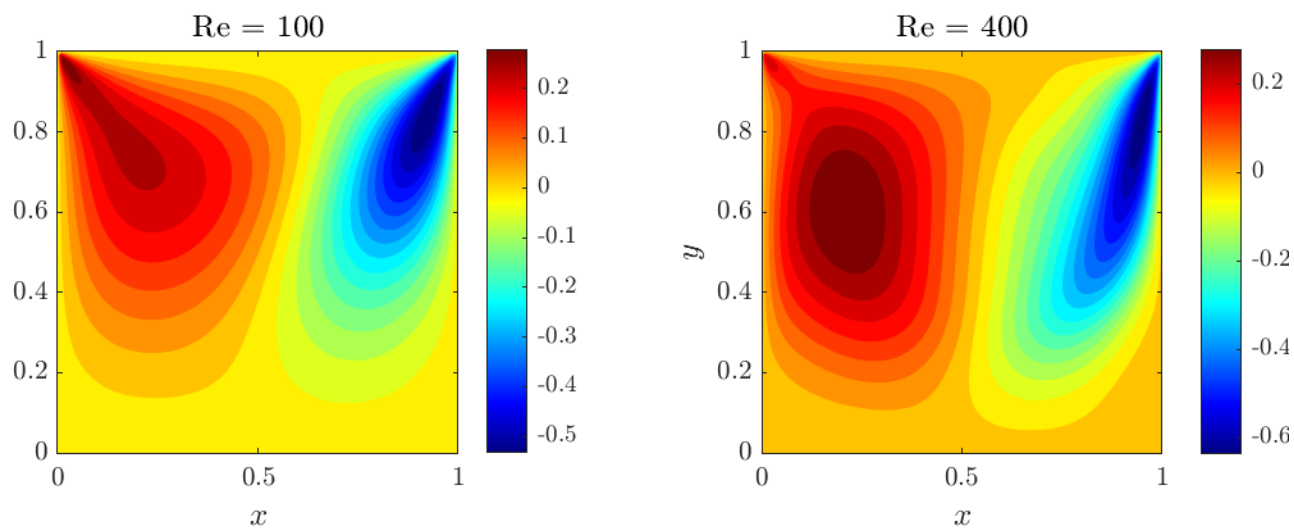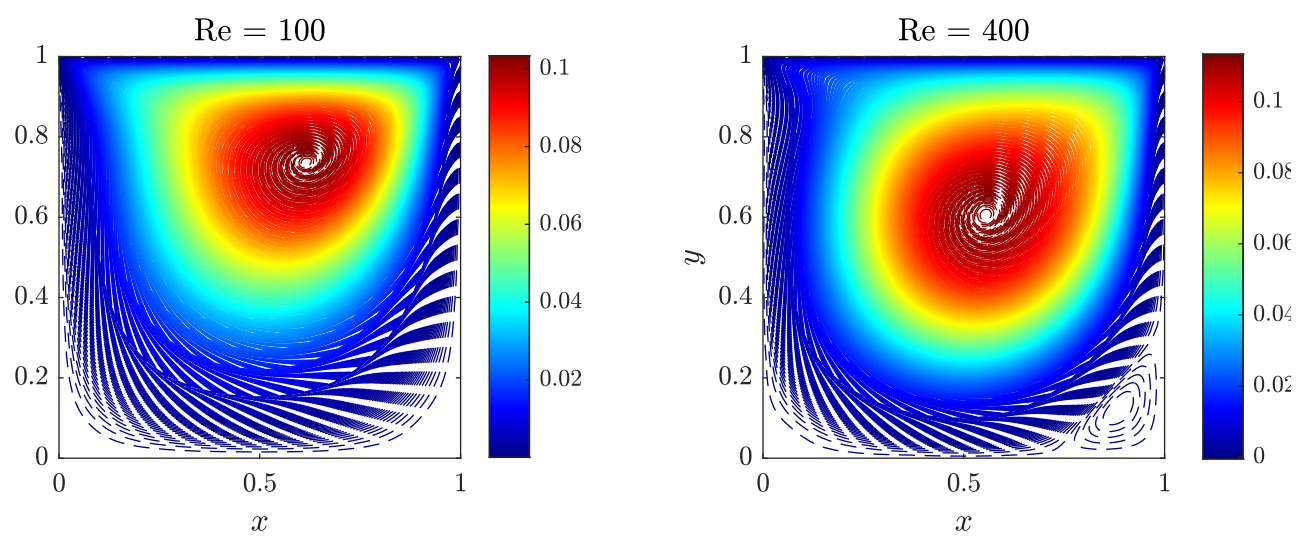
Figure 3.4: v Velocity Contours

Figure 3.5: v Velocity Contours